

# Rajalakshmi Engineering College

Name: Shalini Punithan

Email: 241801258@rajalakshmi.edu.in

Roll no: 241801258

Phone: 8525029597

Branch: REC

Department: I AI & DS FD

Batch: 2028

Degree: B.E - AI & DS

Scan to verify results



## NeoColab\_REC\_CS23231\_DATA STRUCTURES

### REC\_DS using C\_Week 6\_COD\_Question 1

Attempt : 1

Total Mark : 10

Marks Obtained : 0

### Section 1 : Coding

#### 1. Problem Statement

John and Mary are collaborating on a project that involves data analysis. They each have a set of age data, one sorted in ascending order and the other in descending order. However, their analysis requires the data to be in ascending order.

Write a program to help them merge the two sets of age data into a single sorted array in ascending order using merge sort.

#### ***Input Format***

The first line of input consists of an integer N, representing the number of age values in each dataset.

The second line consists of N space-separated integers, representing the ages of participants in John's dataset (in ascending order).

The third line consists of N space-separated integers, representing the ages of participants in Mary's dataset (in descending order).

### **Output Format**

The output prints a single line containing space-separated integers, which represents the merged dataset of ages sorted in ascending order.

Refer to the sample output for formatting specifications.

### **Sample Test Case**

Input: 5

1 3 5 7 9

10 8 6 4 2

Output: 1 2 3 4 5 6 7 8 9 10

### **Answer**

```
#include <stdio.h>
```

```
def merge_sort(a):
```

```
    if len(arr) <= 1:
```

```
        return arr
```

```
    mid = len(arr) // 2
```

```
    left = merge_sort(arr[:mid])
```

```
    right = merge_sort(arr[mid:])
```

```
    result = []
```

```
    i = j = 0
```

```
    while i < len(left) and j < len(right):
```

```
        if left[i] < right[j]:
```

```
            result.append(left[i])
```

```
            i += 1
```

```
        else:
```

```
            result.append(right[j])
```

```
            j += 1
```

```
    result.extend(left[i:])
```

```
    result.extend(right[j:])
```

```
    return result
```

```
N = int(input())
```

```
john = list(map(int, input().split()))
```

```
mary = list(map(int, input().split()))
mary.reverse()
combined = john + mary
sorted_data = merge_sort(combined)
print(' '.join(map(str, sorted_data)))
```

```
int main() {
    int n, m;
    scanf("%d", &n);
    int arr1[n], arr2[n];
    for (int i = 0; i < n; i++) {
        scanf("%d", &arr1[i]);
    }
    for (int i = 0; i < n; i++) {
        scanf("%d", &arr2[i]);
    }
    int merged[n + n];
    mergeSort(arr1, n);
    mergeSort(arr2, n);
    merge(merged, arr1, arr2, n, n);
    for (int i = 0; i < n + n; i++) {
        printf("%d ", merged[i]);
    }
    return 0;
}
```

**Status : Wrong**

**Marks : 0/10**